



AMENDMENT AND RESPONSE UNDER 37 CFR § 1.111

Serial Number: 10/688,573

Filing Date: October 20, 2003

Title: SOFTWARE TOOL FOR SYNTHESIZING A REAL-TIME OPERATING SYSTEM

Page 1  
Dkt: Zeid-01

S/N 10/688,573

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Robert M. Zeidman	Examiner:	Ben C. Wang
Serial No.:	10/688,573	Group Art Unit:	2196
Filed:	October 20, 2003	Docket No.:	Zeid-01
Title:	SOFTWARE TOOL FOR SYNTHESIZING A REAL-TIME OPERATING SYSTEM		

AMENDMENT AND RESPONSE UNDER 37 CFR § 1.116

Mail Stop AF  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

This responds to the Final Office Action mailed on March 13, 2007. Please amend the above-identified patent application as follows to put the case in condition for allowance.

IN THE CLAIMS

Please amend the claims as follows:

- 1) (Currently Amended) A method for developing a real-time operating system, comprising:
  - a) specifying a set of n tasks, task(1) through task(n), to be scheduled for execution, ~~at least one of the tasks of said set of n tasks being selectively configurable as a preemptive or a non-preemptive task;~~
  - b) specifying a scheduling algorithm for scheduling the execution of said set of n tasks; and
  - c) synthesizing source code ~~with embedded~~ from commands embedded in source code to implement a task scheduler that uses said scheduling algorithm ~~and said embedded commands~~ for controlling execution of said set of n tasks, said synthesized source code being executable on a target system after compilation.
- 2) (Currently Amended) The method of claim 1) ~~further~~ including specifying t init-tasks that are executed only once upon initial execution of said task scheduler, t being less than or equal to n.
- 3) (Currently Amended) The method of claim 1) ~~further~~ including specifying f f-loop tasks, each having an associated integer value ~~lc(i)~~ lc(i) for i ranging from 1 to f and f being less than or equal to n, said task scheduler including a continuously executing loop such that each f-loop task executes exactly once every ~~lc(i)~~ lc(i) times that the loop is executed.

5/14/07 BW

Do Not Enter